

**EXAMENSARBETE** Driving Development Resilience:

Analyzing Truck Factors across Proprietary and Open-Source Projects

**STUDENT** Andreas Karlsson**HANDLEDARE** Markus Borg (LTH)**EXAMINATOR** Per Runeson (LTH)

# Att bli påkörd av en buss

## POPULÄRVETENSKAPLIG SAMMANFATTNING **Andreas Karlsson**

Hur skulle din organisation reagera om någon oväntat blev påkörd av en buss? Utöver de personliga tragedierna väcker denna fråga tankar kring hur väl rustad organisationen är för att hantera en plötslig förlust av personal. Har organisationen förmågan att anpassa sig och fortsätta sin verksamhet?

Frågan om vad som händer med en organisation när någon blir påkörd av en buss kan upplevas överdriven. Men trots det så händer det att personal plötsligt lämnar sina roller och sina arbetsplatser. Detta innebär att organisationer behöver ha kunskaper och verktyg för att få en överblick och förståelse i hur beroende verksamheten är av specifika individer.

Inom mjukvaruutveckling så bor mycket av systemets kunskap inom varje utvecklare. Agila arbetsmetoder har premierat mindre dokumentation och mer informella kontaktkanaler vilket har medfört att organisationer har blivit mer beroende av sina utvecklare. Tidigare forskning från öppna källkodsprojekt har identifierat att många system är byggda av relativt få utvecklare. Detta medför att systemen kan mötas av svåra problem om någon/några av dessa utvecklare plötsligt skulle lämna projektet.

Mitt exjobb handlar om att ta fram en algoritm som kan räkna ut Bussfaktorn i ett mjukvaruutvecklingsprojekt. Det vill säga en algoritm som kan identifiera det maximala antal nyckelpersoner som en organisation skulle klara av att plötsligt tappa utan att projektet inte längre skulle kunna

gå att utveckla. Algoritmen är tänkt att användas för att hjälpa organisationer att identifiera vilken risk de har kopplat till plötsligt bortfall av utvecklare.

Den framtagna algoritmen användes också på 195 öppna källkodsprojekt och 102 proprietära projekt för att undersöka om distributionen av Bussfaktorer var liknande mellan de olika kontexterna. Resultatet visade att de olika kontexterna distributioner av Bussfaktorn liknade varandra. Det vill säga, att både öppna källkodsprojekt och proprietära projekt visar liknelser om vilken inneboende risk det finns i fall utvecklare skulle lämna projektet.

Genom den framtagna algoritmen är tanken att företaget CodeScene ska implementera denna i sitt produktutbud. Detta skulle kunna hjälpa CodeScenes kunder att öka sin förståelse och möjlighet att agera på potentiella risker som de eventuellt inte var medvetna om tidigare, och på så sätt minska de negativa påföljderna om någon anställd abrupt skulle lämna organisationen. En konkret åtgärd kan vara att en organisation arbetar mer med att främja kunskapsfördelning för att uppnå ett ökat delat kodägarskap.